Universidad de Carabobo
Facultad Experimental de Ciencias y Tecnología
Departamento de Matemáticas

Quiz Práctico II

Métodos Numéricos II.

1. Considere el sistema de ecuaciones lineales $Ax = b$, donde $A \in M^{n\times n}(\mathbb{R})$, $x, b \in \mathbb{R}^n$. Además, A y b están definidas por:

$$A = \begin{pmatrix} 4 & -1 & 0 & 0 & \cdots & 0 \\ -1 & 4 & -1 & 0 & \cdots & 0 \\ 0 & -1 & 4 & -1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & -1 & 4 & -1 \\ 0 & 0 & \cdots & 0 & -1 & 4 \end{pmatrix}$$

y $b_j = \cos(2\pi j/n)$. Resuelva este sistema para $n = 10$, $n = 100$ y $n = 1000$ usando:

   a) Factorización de Cholesky.

   b) Iteración de Jacobi.

   c) Iteración de Gauss-Seidel.

Elija el residuo relativo $\|x_{n+1} - x_n\|_2/\|x_n\|_2 \leq 10^{-3}$ como criterio de pararda en los métodos iterativos. Elabore un cuadro donde se muestre el tiempo invertido en cada caso. Comente los resultados observados.

2. El perfil de temperaturas $T(x)$ de una barra circular delgada, de radio despreciable y longitud unitaria obedece a la ecuación:

$$\frac{\partial^2 T(x)}{\partial x^2} = f(x)$$

donde $f(x) = 10x^2 + 5x + 10$, $0 \leq x \leq 1$ representa la razón de calor radiante a lo largo de la barra y $T(0) = 300K$ y $T(1) = 400K$ son las conocidas temperaturas al final de la barra. Una solución aproximada de esta ecuación puede ser encontrada usando el método de diferencias finitas. Considere una partición del intervalo en $n+1$ subintervalos con $n+2$ puntos denotados por $x_0 = 0, x_1 = h, \ldots, x_i = ih, \ldots, x_{n+1} = 1$ donde $h(n+1) = 1$ y la temperatura en cada punto es denotada por $T_i = T(x_i)$. El esquema de diferencias finitas es dado por

$$\frac{\partial^2 T(x)}{\partial x^2} \approx \frac{T_{i-1} - 2T_i + T_{i+1}}{h^2}$$

a) Establecer el sistema lineal $Ax = b$ que resulta de utilizar el método de diferencias finitas.

b) Pruebe que la matriz A es diagonal dominante.

c) Resuelva el sistema para obtener soluciones correspondientes a $h = \frac{1}{6}, h = \frac{1}{11}, h = \frac{1}{21}, h = \frac{1}{41}, h = \frac{1}{61}$, usando el método SOR con diferentes valores en el parámetro ω. Establezca, si es posible, una vinculación entre éste y h.

3. En numerosas ocasiones la matriz A del sistema $Ax = b$ es mal condicionada. Una estrategia para resolver estos sistemas es encontrar una matriz P tal que $P^{-1}A$ sea bien condicionada y luego resolver el sistema equivalente

$$P^{-1}Ax = P^{-1}b$$

La matriz P se denomina precondicionador a la izquierda de A. Un requerimiento es que P sea fácil de invertir. Existen numerosos precondicionadores usados en la práctica, de ellos tomar la diagonal de A, es decir:

$$P = D = diag(a_1, \ldots, a_n), \qquad \text{(Precondicionador de Jacobi)}$$

resulta generalmente efectivo cuando A es simétrica y definida positiva y con elementos diagonales no todos iguales. El algoritmo correspondiente al método del Gradiente Conjugado Precondicionado es el siguiente:

x_0: dato inicial

$r_0 = b - Ax_0$

Resolver $Pz_0 = r_0$

$p_0 = z_0$

for $k \leftarrow 0$ **to** N **do**

- $\alpha_k = \dfrac{r_k^t z_k}{p_k^t A p_k}$
- $x_{k+1} = x_k + \alpha_k p_k$
- $r_{k+1} = r_k - \alpha_k A p_k$
- Resolver $Pz_{k+1} = r_{k+1}$
- $\beta_{k+1} = \dfrac{z_{k+1}^t r_{k+1}}{z_k^t r_k}$
- $p_{k+1} = z_{k+1} + \beta_{k+1} p_k$

a) Haga un programa para resolver el problema $Ax = b$ por el método del gradiente conjugado precondicionado con Jacobi.

b) Pruebe su programa con la matriz A definida, para $i, j \in \{1, \ldots, n\}$, por

$$A(i,j) = \begin{cases} i+j & \text{si } 0 < |i-j| < 3 \\ (i+j)^2 & \text{si } i = j \\ 0 & \text{en cualquier otro caso} \end{cases}$$

y $b(i) = i$.

c) Realizar una comparación en número de iteraciones de los métodos Gradiente Conjugado y Gradiente Conjugado precondicionado para la matriz dada considerando distintos valores de n, digamos $n = 5, 10, 20, 30, 50$, etc.

4. Actualmente google se ha establecido como la página de búsqueda en internet más utlizada. Esto ha sido en gran parte gracias al novedoso algoritmo que esta compañia utiliza para, una vez encontradas las páginas que cumplen con el criterio de búsqueda de un usuario, escoger el orden en que éstas serán presentadas. Para escoger este orden google asocia a cada página un número, llamado *pagerank*, que calcula teniendo en cuenta el número de enlaces desde y hacia cada una de las páginas web en el mundo. El orden en que se presentan las páginas depende de su *pagerank* (las de mayor *pagerank* se presentan primero). Para calcular el *pagerank* de cada página web, google crea una matriz G que tiene una fila y una columna por cada página web, las entradas de esta matriz son:

$$G(i,j) = \begin{cases} 1 & \text{si hay un enlace de página } j \text{ a página } i \\ 0 & \text{en caso contrario} \end{cases}$$

El *pagerank* de una página depende no sólo del número de páginas que contengan enlaces a ella, sino también del *pagerank* de esas páginas. Así una página que sea referenciada en muchas páginas de poca importancia podrá tener menos importancia que una que sea referenciada en pocas páginas de mucha importancia. Si x es un vector con el *pagerank* de cada una de las páginas web del mundo, google lo encuentra resolviendo el siguiente sistema de ecuaciones lineales

$$Ax = e, \qquad A = I - pGD \tag{1}$$

dónde I es la matriz de identidad, p es la probabilidad de que una persona, en su búsqueda de alguna información en internet, siga uno de los enlaces en la página donde se encuentra y se toma igual a 0,85, G es la matriz mencionada antes, D es una matriz diagonal tal que $D(i,i)$ es el número de enlaces en la página i y e es el vector que contiene todas sus entradas iguales a 1. Encontremos experimentalmente, de entre todos los algoritmos que hemos estudiado para la solución de sistemas de ecuaciones lineales, cuál es el más conveniente para resolver este sistema de ecuaciones.

a) Guarde en su directorio de trabajo el contenido del archivo `lab2_m.zip`. En él encontrará `data.mat`, `creatematrix_pagerank.m` y `mostrarrangos.m`.

b) Cargue las variables almacenadas en `data.mat` en Matlab. Para ello debe escribir en los comandos de Matlab

$\gg$ `load data.mat`

$\gg$ `whos`

y observe que la matriz G y el vector U se han cargado en memoria (éstas eran las variables guardadas en `data.mat`). La matriz G es como la matriz mencionada antes, pero se creá suponiendo que la web está formada sólo por 500 páginas. La variable U contiene los nombres de estas páginas. Llame ahora a A `=` `creatematrix_pagerank(`G`)` para construir la matriz A en (1).

c) Con `spy(A)` observe la estructura de A. ¿Es una matriz dispersa? ¿Es simétrica? ¿Cuántas de sus entradas son distintas de cero? ¿Será conveniente para google usar un método directo para encontrar el `pagerank` de cada página en internet? ¿Por qué?

d) Busque una aproximación a la solución exacta a este sistema de ecuaciones con los métodos iterativos que usted conoce. ¿Cuáles métodos iterativos le permitieron obtener una aproximación a la solución exacta de (1)? ¿Cuál tolerancia usó? ¿En cuántas iteraciones alcanzó cada uno de ellos la precisión requerida? ¿Cuál es la diferencia entre las soluciones obtenidas? Si alguno de los métodos no le permite obtener una aproximación con la precisión requerida, ¿a qué se debe esto?

e) Una vez encontrada una aproximación a la solución exacta de (1) usted podrá, llamando a la función `mostrarrangos`, ver un código de barras con los rangos de las 10 páginas que tienen mayor *pagerank*, así como una lista con sus direcciones, número de enlaces en y hacia ellas y el *pagerank* de cada una.

NOTA: Se deben tomar a consideración las siguientes observaciones.

- Entregar los archivos .m necesarios para la ejecución correcta de cada pregunta en función de lo exigido.
- Elaborar cada pregunta por separado y se debe indicar cual es el script principal de la pregunta.
- Elaborar un pequeño documento en latex que contenga los resultados teóricos exigidos asi como la definición de las rutinas de cada pregunta, sea organizado.
- Subir al classroom los archivos para su evaluación.